

## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

RECEIVED  
CENTRAL FAX CENTER

NOV 02 2005

In re application of: Bantz et al.

Serial No.: 10/085,547

Filed: February 27, 2002

For: Automatic Provisioning for  
Subscription Computing§  
§  
§  
§  
§  
§

Group Art Unit: 2194

Examiner: Ha, Thanh T.

Attorney Docket No.: YOR920010667US1

54105

PATENT TRADEMARK OFFICE  
CUSTOMER NUMBERCertificate of Transmission Under 37 C.F.R. § 1.8(a)I hereby certify this correspondence is being transmitted via  
facsimile to the Commissioner for Patents, P.O. Box 1450,  
Alexandria, VA 22313-1450, facsimile number (571) 273-8300  
on November 2, 2005.

By:

  
Stephanie FayTRANSMITTAL DOCUMENTCommissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450Sir,  
ENCLOSED HERewith:

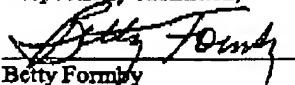
- Appeal Brief (37 C.F.R. 41.37)

RECEIVED  
OIPE/IAP

NOV 03 2005

A fee of \$500.00 is required for filing an Appeal Brief. Please charge this fee to Lenovo (Singapore) Pte. Ltd., a corporation of Singapore, Deposit Account No. 50-3533. No additional fees are believed to be necessary. If, however, any additional fees are required, I authorize the Commissioner to charge these fees which may be required to Lenovo (Singapore) Pte. Ltd., a corporation of Singapore, Deposit Account No. 50-3533. No extension of time is believed to be necessary. If, however, an extension of time is required, the extension is requested, and I authorize the Commissioner to charge any fees for this extension to Lenovo (Singapore) Pte. Ltd., a corporation of Singapore, Deposit Account No. 50-3533.

Respectfully submitted,



Betty Formby

Registration No. 36,536

AGENT FOR APPLICANTS

Duke W. Yee

Registration No. 34,285

ATTORNEY FOR APPLICANTS

YEE &amp; ASSOCIATES, P.C.

P.O. Box 802333

Dallas, Texas 75380

(972) 385-8777

RECEIVED  
CENTRAL FAX CENTER

NOV 02 2005

**Yee &  
Associates, P.C.**4100 Alpha Road  
Suite 1100  
Dallas, Texas 75244Main No. (972) 385-8777  
Facsimile (972) 385-7766**Facsimile Cover Sheet**

To: Commissioner for Patents for Examiner Thanh T. Ha Group Art Unit 2194	Facsimile No.: 571/273-8300
From: Stephanie Fay Legal Assistant to Betty Formby	No. of Pages Including Cover Sheet: 27
Message:  Enclosed herewith: <ul style="list-style-type: none"><li>• Transmittal Document; and</li><li>• Appeal Brief.</li></ul>	
Re: Application No. 10/085,547 Attorney Docket No: YOR920010667US1	
Date: Wednesday, November 02, 2005	
Please contact us at (972) 385-8777 if you do not receive all pages indicated above or experience any difficulty in receiving this facsimile.	<i>This Facsimile is intended only for the use of the addressee and, if the addressee is a client or their agent, contains privileged and confidential information. If you are not the intended recipient of this facsimile, you have received this facsimile inadvertently and in error. Any review, dissemination, distribution, or copying is strictly prohibited. If you received this facsimile in error, please notify us by telephone and return the facsimile to us immediately.</i>

**PLEASE CONFIRM RECEIPT OF THIS TRANSMISSION BY  
FAXING A CONFIRMATION TO 972-385-7766.**

RECEIVED  
CENTRAL FAX CENTER

NOV 02 2005

Docket No. YOR920010667US1

PATENT

## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Bantz et al.

Serial No. 10/085,547

Filed: February 27, 2002

For: Automatic Provisioning for  
Subscription Computing§  
§  
§  
§  
§  
§  
§

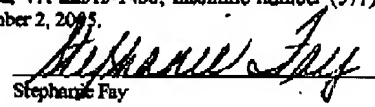
Group Art Unit: 2194

Examiner: Ha, Thanh T.

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450Certificate of Transmission Under 37 C.F.R. § 1.8(a)

I hereby certify this correspondence is being transmitted via facsimile to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, facsimile number (571) 273-8300 on November 2, 2005.

By:

  
Stephanie Fay

## APPEAL BRIEF (37 C.F.R. 41.37)

This brief is in furtherance of the Notice of Appeal, filed in this case on September 14, 2005.

The fees required under § 41.20(B)(2), and any required petition for extension of time for filing this brief and fees therefore, are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

11/03/2005 NGUYEN 00000062 503533 10085547

01 FC:1402 500.00 DA

(Appeal Brief Page 1 of 25)  
Bantz et al. - 10/085,547

**REAL PARTY IN INTEREST**

The real party in interest in this appeal is the following party: Lenovo (Singapore) Pte. Ltd., a corporation of Singapore, having a place of business at 9 Changi Business Park, Central 1, Singapore 486048.

**RELATED APPEALS AND INTERFERENCES**

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

**STATUS OF CLAIMS****A. TOTAL NUMBER OF CLAIMS IN APPLICATION**

Claims in the application are: 1-50

**B. STATUS OF ALL THE CLAIMS IN APPLICATION**

1. Claims canceled: 3 and 48
2. Claims withdrawn from consideration but not canceled: None
3. Claims pending: 1-2, 4-47, and 49-50
4. Claims allowed: None
5. Claims rejected: 1-2, 4-47, and 49-50
6. Claims objected to: None

**C. CLAIMS ON APPEAL**

The claims on appeal are: 1-2, 4-47, and 49-50

**STATUS OF AMENDMENTS**

No amendments have been submitted since the final rejection.

### **SUMMARY OF CLAIMED SUBJECT MATTER**

#### **A. CLAIM 1 - INDEPENDENT**

The subject matter of claim 1 is directed to a method for creating customized disk images for loading software onto a computer. The method is shown in **Figures 5-7**, discussed on page 11, line 12 through page 13, line 17. The method contains the following steps:

- receiving software requirements for a given system from a plurality of users – This step is shown in **Figure 5**, step 200, discussed on page 11, lines 14-16. Additionally, **Figure 2A**, discussed on page 7, line 22 through page 8, line 15, illustrates that the requirements from a system are received from a number of different users.
- determining (a) a plurality of software components that will fulfill the software requirements while addressing constraints and affinities between said plurality of software components and (b) a respective plurality of configuration options that reflect current best practices with regard to said plurality of software components – This step is shown as steps 203-204 of **Figure 5**, and is further elaborated in **Figure 6**, discussed on page 12, line 6-24.
- generating a disk image containing said plurality of software components configured according to said respective plurality of configuration options – This step is shown in **Figure 7**, especially in step 243; the flowchart of **Figure 7** is discussed on page 12, line 25 through page 13, line 17.

#### **B. CLAIM 15 - DEPENDENT**

The subject matter of claim 15 is directed to clarifying that the software requirements are received in a structured format that is eXtensible Markup Language (XML).

#### **C. CLAIM 16 - INDEPENDENT**

The subject matter of claim 16 is directed to a method for creating a customized disk image for loading software onto a computer. The method is shown in **Figures 5-7**, discussed on page 11, line 12 through page 13, line 17. The method contains the following steps:



- parsing a plurality of inputs for a desired system to extract specifications regarding software. This step is shown in Figure 5 as steps 201 and 202, discussed on page 11, lines 17-22.
- evaluating rules that apply to the plurality of inputs to derive a set of software components conforming to the specifications;
- evaluating additional rules that apply to the plurality of inputs to derive a set of configuration options conforming to the specifications. Both evaluating steps are shown as step 203 in Figure 5, discussed on page 11, lines 23-24. Exemplary embodiments of the rules and configurations are shown as contained in knowledge bases 91, 92, and 93, discussed on page 10, lines 5-18.
- storing each software component from the set of software components on a storage device;
- configuring each software component stored on the storage device in accordance to the set of configuration options. These two steps are shown as step 204 of Figure 5, discussed on page 11, lines 23-24, and as step 226 of Figure 6, discussed on page 12, lines 19-20, which show generating the provisioning order.
- generating a disk image from contents of the storage device. This step is shown in Figure 7, especially step 243; the step is discussed on page 12, line 25 through page 13, line 13.

#### **D. CLAIM 17 - DEPENDENT**

The subject matter of claim 17 is directed to further clarifying the method of claim 16 and defines the inputs as requests from hypertext browsers. This feature is discussed on page 8, lines 9-10.

#### **E. CLAIM 19 - INDEPENDENT**

The subject matter of claim 19 is directed to a computer program product and corresponds to claim 1. The computer program product is discussed on page 16, lines 13-24.

**F. CLAIM 34 - INDEPENDENT**

The subject matter of claim 34 is directed to a data processing system and corresponds to claim

1. The data processing system is depicted in Figure 3, discussed on page 9, line 23 through page 10, line 23.

**GROUND OF REJECTION TO BE REVIEWED ON APPEAL**

**A. GROUND OF REJECTION 1 (Claims 1-2, 4-14, 16, 19-32, 34-47, 49-50)**

Claims 1-2, 4-14, 16, 19-32, 34-47, 49-50 stand rejected under 35 U.S.C. § 103(a) as obvious over Kroening, Method for Configuring Software for a Build to Order System, U.S. Patent No. 6,775,829, August 10, 2004<sup>1</sup> (hereinafter "Kroening") in view of Marino *et al.*, Method and System for Installing Software on a Computer System, U.S. Patent No. 6,681,391, January 20, 2004 (hereinafter "Marino").

**B. GROUND OF REJECTION 2 (Claims 15 and 33)**

Claims 15 and 33 stand rejected under 35 U.S.C. § 103(a) as obvious over Kroening in view of Marino, further in view of Humpleman *et al.*, Method and Apparatus for User and Device Command and Control in a Network, U.S. Patent No. 6,546,419, April 8, 2003<sup>2</sup> (hereinafter "Humpleman").

**C. GROUND OF REJECTION 3 (Claims 17-18)**

Claims 17-18 stand rejected under 35 U.S.C. § 103(a) as obvious over Kroening, Marino, further in view of Lo *et al.*, Accessing a ERP Application Over the Internet Using Strongly Typed Declarative Language Files, U.S. Patent No. 6,854,120, February 8, 2005 (hereinafter "Lo").

<sup>1</sup> This patent is subject to a terminal disclaimer.

<sup>2</sup> This patent is subject to a terminal disclaimer.

### ARGUMENTS

#### A. GROUND OF REJECTION 1 (Claims 1-2, 4-14, 16, 19-32, 34-47, 49-50)

The rejection states:

As Per Claim 1, Kroening teaches a method for creating customized disk images for loading software onto a computer (E.g. see Abstract and associated text), the method comprising the steps:

Receiving software requirements (E.g. see FIG. 2, order 208 and associated text) from a plurality of users (E.g. see FIG. 2, user 202, 204 206 and associated text & col. 11, lines 7-8);

Determining (a) a plurality of software components that will fulfill the software requirements while addressing constraints and affinities between said plurality of software components and (b) a respective plurality of configuration options that reflect current best practices with regard to said plurality of software components (col. 6, lines 17-27); and

generating a disk image containing said plurality software components configured according to said respective plurality of configuration options. (E.g. see FIG. 2 and associated text, e.g. see col. 5:59 to col. 6:50).

Kroening does not specifically teach software requirements for a given system. However, Marino teaches the software requirements for a given system from a plurality of users ("many different user") [col. 2, lines 19-51].

It would have been obvious to one of ordinary skill in the art at the time the invention was made to have combined the teaching of Marino and Kroening because Marino's teaching would customized the installation software image for a plurality of users without requiring expensive oversight or trouble-shooting to install the software image.

If the Patent Office does not produce a *prima facie* case of unpatentability, then without more the applicant is entitled to grant of a patent. *In re Oetiker*, 977 F.2d 1443, 1445, 24 U.S.P.Q.2d 1443, 1444 (Fed. Cir. 1992); *In re Grabiak*, 769 F.2d 729, 733, 226 U.S.P.Q. 870, 873 (Fed. Cir. 1985). A *prima facie* case of obviousness is established when the teachings of the prior art itself suggest the claimed subject matter to a person of ordinary skill in the art. *In re Bell*, 991 F.2d 781, 783, 26 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1993). The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990).

Exemplary claim 1 is not obvious over Kroening and Marino for at least two reasons:

(a) the combination of these references would not meet the feature of "receiving software

requirements for a given system from a plurality of users" and (b) one of ordinary skill in the art would not use the suggested motivation to combine these references. Each will be discussed separately.

**Claim feature: receiving requirements from a plurality of users**

In the invention recited in claim 1, software requirements are received for "a given system from a plurality of users". This feature reflects the needs of a business in which input is desired from a number of people who understand different aspects of the business. Rather than require an administrator to collect and collate these different requirements, the methodology of claim 1 allows the system to collect the requirements from the different users.

In contrast, although Kroening receives input from multiple users, each user is creating their own system, rather than each providing input for the same system. The exemplary embodiment of Kroening states,

In referring to FIG. 2, an exemplary embodiment 200 is shown wherein a logic flow for creating a disk image of a desired software configuration is illustrated. A plurality of users 202, 204 and 206 enter orders for a first information handling system, a second information handling system and up to an "Nth" information handling system. The first information handling system is different from the second information handling system.<sup>3</sup>

One of ordinary skill in the art would understand from this excerpt and from Kroening as a whole that the system of Kroening takes each user's input and creates a corresponding information handling system, but does not use multiple user input for a single system.

Additionally, contrary to statements in the rejection, Marino does not disclose multiple users providing input for a given system. The rejection quoted above asserts that "*Marino teaches the software requirements for a given system from a plurality of users ('many different user') [col. 2, lines 19-51].*" However, when one looks at the entire thought in Marino, this patent states, "*Furthermore, determination of an installation order in accordance with the invention allows many different user configurations to be detected and modified as required without requiring expensive oversight or trouble-shooting to install a suite of interest, including via automated installations.*"<sup>4</sup> Marino recognizes that there can be many user configurations, but this does not translate into many users providing input to a system. Thus, the step of "*receiving*

<sup>3</sup> Kroening, column 5, lines 61-64

<sup>4</sup> Marino, column 2, lines 46-51

*software requirements for a given system from a plurality of users*" is not met by either Kroening or Marino, nor by their combination.

**Motive to Combine**

The rejection suggests, as the motive for combining Marino with Kroening, that:

It would have been obvious to one of ordinary skill in the art at the time the invention was made to have combined the teaching of Marino and Kroening because Marino's teaching would customized the installation software image for a plurality of users without requiring expensive oversight or trouble-shooting to install the software image.

However, this motivation would not have been used by one of ordinary skill in the art as a reason to combine these patents. Kroening notes that it provides a "*custom software configuration*"<sup>5</sup> for installation on a "*build-to-order system*"<sup>6</sup>. It discloses this capability as already existing in the system it discloses, which provides the customized service for new systems. Therefore, there would be no motivation to provide an additional customization from Marino, as the customization already exists. Marino does also offer this customization for upgrading existing suites, but since Kroening is directed solely to customizing new systems, there is again no motivation to combine these patents. Thus, the cited motivation to combine does not appear to be one that would be used by one of ordinary skill in the art,

Therefore, claim 1 and the other claims in this grouping have been shown to be allowable over the combination of Kroening with Marino.

**B. GROUND OF REJECTION 2 (Claims 15 and 33)**

The rejection states:

As Per Claim 15, the rejection of claim 14 is incorporated and further Kroening and Marino do not explicitly disclose Extensible Markup Language (XML). However, Humpleman in an analogous art teaches wherein the structured format is Extensible Markup Language (.XML)". (E.g. see col. 3:10-17). Therefore, it would have been obvious to incorporate the teaching of Humpleman into the teaching of Kroening to use the structured format like XML. The modification would have been obvious because one of ordinary skill in the art would have been motivated so that the selection information for each home device can include an iconic representation of the home device. And, the structured format can include the XML format.

<sup>5</sup> Kroening, abstract

<sup>6</sup> Kroening, title

A prima facie case of obviousness is established when the teachings of the prior art itself suggest the claimed subject matter to a person of ordinary skill in the art. *In re Bell*, 991 F.2d 781, 783, 26 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1993). The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990). A prior art reference must be considered in its entirety, i.e., as a whole, including portions that would lead away from the claimed invention. *W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 220 USPQ 303 (Fed. Cir. 1983), *cert. denied*, 469 U.S. 851 (1984).

Representative claim 15 is dependent on claim 14, which recites “*wherein the requirements are represented in a structured format*”; claim 15 recites, “*wherein the structured format is Extensible Markup Language (XML)*”.

This claim is not obvious over the combination of **Kroening**, **Marino**, and **Humpleman** because one of ordinary skill in the art would not combine **Humpleman** with the others when they are considered as a whole. In considering the references as a whole, one of ordinary skill in the art would look at the problems recognized and solved. **Kroening** and **Marino** are generally directed towards configuring and loading software onto computers, as shown by their titles: Method for Configuring Software for a Build to Order System (**Kroening**) and Method and System for Installing Software on a Computer System (**Marino**). In contrast, **Humpleman** recognizes problems with “*network systems, and more particularly, ... [with a] home network having multiple devices connected thereto*”<sup>7</sup>. These two problems are unrelated and one of ordinary skill in the art would not be motivated to combine these references when they are read a whole. As further support, the cited references provide entirely different solutions: Although **Kroening** and **Marino** differ in their methodology, both recite either “*configuring software*” (**Kroening**) or “*installing a software suite*” (**Marino**) on a computer. In contrast, **Humpleman** provides “*method and system for performing a service on a home network having a plurality of home devices connected thereto*”. Thus, one of ordinary skill in the art would not be motivated to combine these references in the manner suggested by the examiner. The references can be combined only through the improper use of hindsight with the benefit of applicants’ disclosure as a template to reach the presently claimed invention.

---

<sup>7</sup> Humpleman, column 1, lines 28-30

**C. GROUND OF REJECTION 3 (Claims 17-18)**

The rejection states:

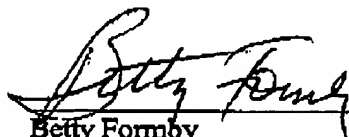
As Per claim 17, the rejection of claim 16 is incorporated and further Kroening does not explicitly disclose inputs are requests from hypertext browsers. However, Lo in an analogous art teaches wherein the plurality of inputs are requests form hypertext browsers (E.g. see col. 2:21-47). Therefore, it would have been obvious to incorporate the teaching of Lo into the teaching of Kroening so that inputs are requests from hypertext browsers. The modification would have been obvious because one of ordinary skill in the art would have been motivated to transfer any data entered by the user into the HTML input form and any data stored in the requested HTML page into the ERP application API (E.g. see col. 2:33-35).

A prior art reference must be considered in its entirety, i.e., as a whole, including portions that would lead away from the claimed invention. *W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 220 USPQ 303 (Fed. Cir. 1983), *cert. denied*, 469 U.S. 851 (1984).

Representative 17 is dependent on claim 16 and recites "*wherein the plurality of inputs are requests from hypertext browsers*". This claim is not obvious over the combination of **Kroening**, **Marino**, and **Lo** because one of ordinary skill in the art would not combine **Lo** with the other patents when they are considered as a whole. In considering the references as a whole, one of ordinary skill in the art would look at the problems recognized and solved. **Kroening** and **Marino** are generally directed towards configuring and loading software onto computers, as noted above. In contrast, **Lo** recognizes problems with "*enterprise resource planning systems performed by computers ... [especially] accessing an enterprise resource planning application over the Internet using Java*". These two problems are unrelated and one of ordinary skill in the art would not be motivated to combine these references when they are read a whole. As further support, the cited references provide entirely different solutions: **Kroening** and **Marino** prepare software for installation on a system. In contrast, **Lo** provides "*a method and apparatus for executing pre-defined API calls in an ERP system via the Internet*". Thus, one of ordinary skill in the art would not be motivated to combine these references in the manner suggested by the examiner. The references can be combined only through the improper use of hindsight with the benefit of applicants' disclosure as a template to reach the presently claimed invention.



The Board of Appeals is requested to reverse the outstanding rejections and indicate all claims to be allowable.



Betty Formby

Reg. No. 36,536

YEE & ASSOCIATES, P.C.

PO Box 802333

Dallas, TX 75380

(972) 385-8777

**CLAIMS APPENDIX**

The text of the claims involved in the appeal are:

1. A method for creating customized disk images for loading software onto a computer, the method comprising the steps:

receiving software requirements for a given system from a plurality of users;

determining (a) a plurality of software components that will fulfill the software requirements while addressing constraints and affinities between said plurality of software components and (b) a respective plurality of configuration options that reflect current best practices with regard to said plurality of software components; and

generating a disk image containing said plurality of software components configured according to said respective plurality of configuration options.

2. The method of claim 1, wherein said determining step applies rules to the software requirements to identify software components that comply with the software requirements.
3. (Cancelled)
4. The method of claim 2, wherein the rules include rules mapping a software requirement into a corresponding software component.
5. The method of claim 2, wherein the rules include rules specifying when particular versions of a particular software component are to be utilized.

6. The method of claim 2, wherein the rules include rules specifying installation options regarding a particular software component.
7. The method of claim 2, wherein the rules include rules specifying how to test a particular software component.
8. The method of claim 1, further comprising:  
testing the disk image.
9. The method of claim 8, wherein testing the disk image includes verifying that said plurality of software components complies with the software requirements.
10. The method of claim 8, wherein testing the disk image includes verifying that said plurality of software components complies with at least one rule.
11. The method of claim 1, further comprising:  
generating a difference image that represents differences between the disk image and another existing disk image, whereby the another existing disk image may be updated to match the disk image by applying the difference image to the another existing disk image.
12. The method of claim 1, wherein the software requirements are received through a network that includes the Internet.
13. The method of claim 1, wherein the software requirements can be received in terms of customer needs rather than specific software components.
14. The method of claim 1, wherein the requirements are represented in a structured format.

15. The method of claim 14, wherein the structured format is Extensible Markup Language (XML).

16. A method for creating a customized disk image for loading software onto a computer, the method comprising the computer-implemented steps:

    parsing a plurality of inputs regarding a desired system to extract specifications regarding software;

    evaluating a plurality of rules with respect to the plurality of inputs to derive a set of software components conforming to the specifications;

    evaluating a second plurality of rules with respect to the plurality of inputs to derive a set of configuration options conforming to at least the specifications;

    storing each software component from the set of software components on a storage device;

    configuring each software component stored on the storage device in accordance to the set of configuration options; and

    generating a disk image from contents of the storage device.

17. The method of claim 16, wherein the plurality of inputs are requests from hypertext browsers.

18. The method of claim 16, wherein the plurality of inputs are XML documents.

19. A computer program product stored in a computer-readable medium and comprising functional descriptive data that, when executed by a computer, enables the computer to create customized disk images for loading software onto a computer, including the steps:

receiving software requirements for a given system from a plurality of users;  
determining (a) a plurality of software components that will fulfill the software requirements while addressing constraints and affinities between said plurality of software components and (b) a respective plurality of configuration options that reflect current best practices with regard to said plurality of software components; and  
generating a disk image containing said plurality of software components configured according to said respective plurality of configuration options.

20. The computer program product of claim 19, wherein said determining step applies rules to the software requirements to identify software component that comply with the software requirements.
21. The computer program product of claim 20, wherein the rules are stored in a database.
22. The computer program product of claim 21, wherein the rules include rules mapping a software requirement into a corresponding software component.
23. The computer program product of claim 21, wherein the rules include rules specifying when particular versions of a particular software component are to be utilized.
24. The computer program product of claim 21, wherein the rules include rules specifying installation options regarding a particular software component.
25. The computer program product of claim 21, wherein the rules include rules specifying how to test a particular software component.

26. The computer program product of claim 19, comprising additional functional descriptive data that, when executed by the computer, enables the computer to perform additional acts including:

testing the disk image.

27. The computer program product of claim 26, wherein testing the disk image includes verifying that said plurality of software components complies with the software requirements.

28. The computer program product of claim 26, wherein testing the disk image includes verifying that said plurality of software components complies with at least one rule.

29. The computer program product of claim 19, comprising additional functional descriptive data that, when executed by the computer, enables the computer to perform additional acts including:

generating a difference image that represents differences between the disk image and another existing disk image, whereby the another existing disk image may be updated to match the disk image by applying the difference image to the another existing disk image.

30. The computer program product of claim 19, wherein the software requirements are received through a network that includes the Internet.

31. The computer program product of claim 19, wherein the software requirements can be received in terms of customer needs rather than specific software components.

32. The computer program product of claim 19, wherein the requirements are represented in a structured format.

33. The computer program product of claim 32, wherein the structured format is Extensible Markup Language (XML).

34. A data processing system capable of creating customized disk images for loading software onto a computer, said data processing system comprising:

instructions for receiving software requirements for a given system from a plurality of users;

instructions for determining (a) a plurality of software components that will fulfill the software requirements while addressing constraints and affinities between said plurality of software components and (b) a respective plurality of configuration options that reflect current best practices with regard to said plurality of software components; and

instructions for generating a disk image containing said plurality of software components configured according to said respective plurality of configuration options.

35. The data processing system of claim 34, wherein said instructions for determining applies rules to the software requirements to identify software component that comply with the software requirements.

36. The data processing system of claim 35, wherein the rules are stored in a database.

37. The data processing system of claim 36, wherein the rules include rules mapping a software requirement into a corresponding software component.

38. The data processing system of claim 36, wherein the rules include rules specifying when particular versions of a particular software component are to be utilized.

39. The data processing system of claim 36, wherein the rules include rules specifying installation options regarding a particular software component.
40. The data processing system of claim 36, wherein the rules include rules specifying how to test a particular software component.
41. The data processing system of claim 34, further comprising:  
means for testing the disk image.
42. The data processing system of claim 41, wherein testing the disk image includes verifying that said plurality of software components complies with the software requirements.
43. The data processing system of claim 41, wherein testing the disk image includes verifying that item said plurality of software complies with at least one rule.
44. The data processing system of claim 34, further comprising:  
means for generating a difference image that represents differences between the disk image and another existing disk image, whereby the another existing disk image may be updated to match the disk image by applying the difference image to the another existing disk image.
45. The data processing system of claim 34, wherein the software requirements are received through a network that includes the Internet.
46. The data processing system of claim 34, wherein the software requirements can be received in terms of customer needs rather than specific software components.
47. The data processing system of claim 34, wherein the requirements are represented in a structured format, such as Extensible Markup Language (XML).



48. (Cancelled)

49. The method of claim 1, further comprising storing said disk image on a computer-readable media and distributing said computer-readable media to a client.

50. The data processing system of claim 34, further comprising instructions for storing said disk image on a computer-readable media, wherein said computer-readable media can be distributed to a client.

**EVIDENCE APPENDIX**

There is no evidence to be presented.

**RELATED PROCEEDINGS APPENDIX**

There are no related proceedings.